Introduction to Python I (Exercises 04) Functions

1) Write a function called message that prints the following text:

   "This is text printed inside a function!"

   (Notice that there are no params and no returns)

```
def message():
    print("This is text printed inside a function! ")


#call the function message

message()
```

2) Write a function that receives a float number indicating the radius of a circle.

   The function should calculate the area of the circle according to the following equation:

   area = PI x radius^2 (you can approximate Pi to 3.1415)

   Return the area to the calling section of the program. Test it.

```
def circle_area(radius):
    area = 3.1415 * radius**2

    return area


#call the function message and use the returned value in a print
# statement

radius = 10.5

print("The area of the circle of radius: " , radius, "is: ", circle_area(radius))
```

3) Write a function “rectangle” that receives two integer numbers indicating Height and Width. Inside the function filter the values of height to be between 3 and 10 (if it is not one of these values, then make the height equal to 3. Filter the width value to be between 3 and 10 (if it is not one of these values, then make the width equal to 3).
The function should then print a rectangle with the character *

```
def rectangle(width, height):
    if width < 3 or width > 10:
        width = 3
    if height < 3 or height > 10:
        height = 3

    for i in range(height):
        print ('*' * width)

#start of program
input_width = int(input("Enter the width of the rectangle: "))
input_height = int(input("Enter the height of the rectangle: "))

#draw the rectangle by calling the "rectangle()" function defined above

rectangle(input_width, input_height)
```

4) Write a function that receives 3 parameters. The first parameter is a positional parameter (it is mandatory to pass the parameter). The other two parameters contain default values (The default values are 1 and 10 respectively).

As in: def myfunction(myvalue, defvalue1 = 1, defvalue2=10):
totalvalue = (myvalue * defvalue2) + 1
return totalvalue

Test you function for each of the following cases:

4.1 myfunction(1)
myfunction(10)
myfunction(1,1)
myfunction(1,1,5)
myfunction(10,defvalue2 = 5, defvalue1=10)
myfunction(10,defvalue2 = 5)

What happens in these cases?

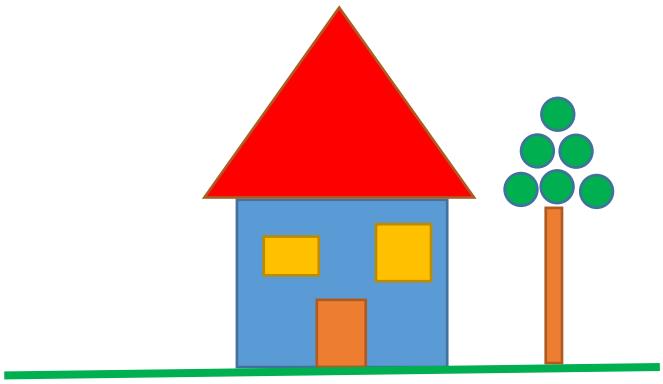
myfunction(defvalue1 = 2)

```
myfunction(defvalue1 = 2, defvalue2 = 10)
myfunction()
myfunction(1, defvalue3 = 5)
```

5) Using Turtle Graphics, write the following functions:

. drawline(x1,y1,x2,y2,color='black')
. drawsquare (height, width, color='black',fill='white', x,y)
.drawtriangle(height, width, color='black',fill='white', x,y)
.drawcircle(radius, color='black', fill='white')

Test your functions by making a drawing similar to the following:



Potential functions (try them). See how the parameters are passed, and how we use default parameters (this is just a sample of possible functions. They can be enhanced easily with a bit more time.

```
from turtle import *

def drawline(x1,y1,x2,y2,lcolor='black'):
    '''
       Draws a line from (x1,y1) to (x2,y2)
       using the specified turtle and the
       defaul color of 'black'. The color can be change
       by assigning a value to the keyword parameter color.
    '''
    color(lcolor)
    up()
    goto(x1,y1)
    down()
    goto(x2,y2)
    up()
```

```
def drawrectangle(height,width,x,y,lcolor='black',fill='white'):
    '''
       Draws a rectangle of with the given height and width
       starting and position(x,y)
       using the specified turtle and the
       default color of 'black' and fill of ' white'
    '''
        color(lcolor)
    fillcolor(fill)
    begin_fill()
    up()
    goto(x,y)
    down()
    forward(width)
    right(90)
    forward(height)
    right(90)
    forward(width)
    right(90)
    forward(height)
    right(90)
    end_fill()
    up()

def drawtriangle(height, width, x,y,lcolor='black',fill='white'):
    '''
       drawtriangle draws an equilateral triangle according to Heigt and
       Witdh parameters. It uses the drawline() function to achieve this.

    '''
    color(lcolor)
    begin_fill()
    up()
    goto(x,y)
    down()
    drawline(x,y, x+width, y, lcolor=lcolor)
    drawline(x+width,y,x+(width/2),y+height,lcolor=lcolor)
    drawline(x+(width/2),y+height, x,y,lcolor=lcolor)
    fillcolor(fill)
    end_fill()
    up()

def drawcircle(radius,x,y,lcolor='black', fill='white'):

    color(lcolor)
    fillcolor(fill)
    begin_fill()
    up()
    goto(x,y)
    down()
    circle(radius)
    end_fill()
    up()
```

```
# Start of program logic

# Draw the ground
drawrectangle(10,300,-100,-100,fill='green')

#Draw the Main Frame of the house
drawrectangle(100,150,-15,0,fill="cyan")

# Draw the ceiling
drawtriangle(100,200,-40,0, fill='Red')

# The door
drawrectangle(40,20,0,-60,fill='brown')

# The windows
drawrectangle(15,30,0,-30,)
drawrectangle(25,60,60,-45)

# The tree Trunk
drawrectangle(60,10,175,-40, fill='brown')

# The leaves
drawcircle(15,160,-50,fill='green')
drawcircle(15,180,-50,fill='green')
drawcircle(15,200,-50,fill='green')
drawcircle(13,175,-35,fill='green')
drawcircle(13,195,-35,fill='green')
drawcircle(10,180,-15,fill='green')

#turtle.update()
exitonclick()
```